

Package ‘pcutils’

June 13, 2023

Type Package

Title Some Useful Functions for Statistics and Visualization

Version 0.1.0

Author Peng Chen [aut, cre]

Maintainer Peng Chen <pengchen2001@zju.edu.cn>

Description Offers a range of utilities and functions for everyday programming tasks.

1.Data Manipulation. Such as grouping and merging, column splitting, and character expansion.

2.File Handling. Read and convert files in popular formats, including ``blast'', ``diamond'', ``fasta'', ``gff'', ``gtf'', and various image formats like ``jpg'', ``png'', ``pdf'', and ``svg''.

3.Plotting Assistance. Helpful utilities for generating color palettes, validating color formats, and adding transparency.

4.Statistical Analysis. Includes functions for pairwise comparisons and multiple testing corrections,

enabling perform statistical analyses with ease.

5.Graph Plotting, Provides efficient tools for creating doughnut plot and multi-layered doughnut plot;

Venn diagrams, including traditional Venn diagrams, upset plots, and flower plots;

Simplified functions for creating stacked bar plots, or a box plot with alphabets group for multiple comparison group.

License GPL-3

Encoding UTF-8

RoxygenNote 7.2.3

Depends R (>= 4.0.0)

Imports dplyr, magrittr, ggplot2, stats, utils, grDevices, reshape2, scales, agricolae,

Suggests testthat, clipr, remotes, BiocManager, ggpubr, kableExtra, RImagePalette, animation, htmlwidgets, jpeg, pagedown, png, tidyr, RColorBrewer, ggsci, readr, tools, graphics, grImport2, pdftools, rsvg, PMCMRplus, nortest, fitdistrplus, ggalluvial, gghalves, pheatmap, corrplot, ggspatial, sf, ggimage, ggpmisc, ggvenn, UpSetR, plotrix, RBGL, graph, ggtern, wordcloud2, circlize, RIdeogram, igraph, tibble, knitr, rmarkdown, prettydoc

NeedsCompilation no

Repository CRAN

Date/Publication 2023-06-13 08:00:02 UTC

R topics documented:

add_alpha	3
add_analysis	3
add_theme	4
change_fac_lev	5
china_map	5
copy_vector	6
count2	6
dabiao	7
del_ps	7
explode	8
fittest	8
get_cols	9
get_doi	9
gghuan	10
gghuan2	11
grepl.data.frame	12
group_box	12
hebing	14
how_to_update_parameters	14
how_to_use_parallel	15
is.ggplot.color	15
legend_size	16
lib_ps	16
little_guodong	17
make_project	17
metadata	18
mmscale	18
multitest	19
my_cat	19
my_circo	20
my_lm	20
otutab	21
plotgif	21
plotpdf	22
read.file	23
read_fasta	23
remove.outliers	24
rgb2code	24
sanxian	25
stackplot	25
strsplit2	27

<i>add_alpha</i>	3
taxonomy	28
tax_pie	28
tax_wordcloud	29
trans_format	29
twotest	30
update_param	31
venn	31
write_fasta	32
Index	33

<code>add_alpha</code>	<i>Add alpha for a Rcolor</i>
------------------------	-------------------------------

Description

Add alpha for a Rcolor

Usage

```
add_alpha(color, alpha = 0.3)
```

Arguments

<code>color</code>	Rcolor
<code>alpha</code>	alpha, default 0.3

Value

8 hex color

Examples

```
add_alpha("red",0.3)
```

<code>add_analysis</code>	<i>Add an analysis for a project</i>
---------------------------	--------------------------------------

Description

Add an analysis for a project

Usage

```
add_analysis(analysis_n, title = analysis_n, author = "Peng Chen", theme = 1)
```

Arguments

analysis_n	analysis name
title	file title
author	author
theme	1~10

Value

No return value

add_theme	<i>Add a global gg_theme and colors for plots</i>
-----------	---

Description

Add a global gg_theme and colors for plots

Usage

```
add_theme(set_theme = NULL)
```

Arguments

set_theme	your theme
-----------	------------

Value

No return value

Examples

```
add_theme()
```

change_fac_lev	<i>Change factor levels</i>
----------------	-----------------------------

Description

Change factor levels

Usage

```
change_fac_lev(x, levels = NULL)
```

Arguments

x	vector
levels	custom levels

Value

factor

Examples

```
change_fac_lev(letters[1:5], levels = c("c", "a"))
```

china_map	<i>Plot china map</i>
-----------	-----------------------

Description

Plot china map

Usage

```
china_map(dir = "~/database/")
```

Arguments

dir	where to put the china.json file
-----	----------------------------------

Value

a ggplot

copy_vector	<i>Copy a vector</i>
-------------	----------------------

Description

Copy a vector

Usage

```
copy_vector(vec)
```

Arguments

vec a R vector object

Value

No return value

count2	<i>Like uniq -c in shell to count a vector</i>
--------	--

Description

Like uniq -c in shell to count a vector

Usage

```
count2(df)
```

Arguments

df two columns: first is type, second is number

Value

two columns: first is type, second is number

Examples

```
count2(data.frame(group = c("A", "A", "B", "C", "C", "A"), value = c(2, 2, 2, 1, 3, 1)))
```

dabiao *Print some message with =*

Description

Print some message with =

Usage

```
dabiao(str = "", ..., n = 80, char = "=", mode = c("middle", "left", "right"))
```

Arguments

str	output strings
...	strings will be paste together
n	the number of output length
char	side chars default:=
mode	"middle", "left" or "right"

Value

No return value

Examples

```
dabiao("Start running!")
```

del_ps *Detach packages*

Description

Detach packages

Usage

```
del_ps(p_list, ..., origin = NULL)
```

Arguments

p_list	a vector of packages list
...	packages
origin	keep the original Namespace

Value

No return value

explode	<i>Explode a data.frame if there are split charter in one column</i>
---------	--

Description

Explode a data.frame if there are split charter in one column

Usage

```
explode(df, column, split = ",")
```

Arguments

df	data.frame
column	column
split	split string

Value

data.frame

Examples

```
df <- data.frame(a = 1:2, b = c("a,b", "c"), c = 3:4)
explode(df, "b", ",")
```

fittest	<i>Fit a distribution</i>
---------	---------------------------

Description

Fit a distribution

Usage

```
fittest(a)
```

Arguments

a	a numeric vector
---	------------------

Value

distribution

get_cols	<i>Get n colors</i>
----------	---------------------

Description

Get n colors

Usage

```
get_cols(n, pal = "col1", picture = NULL)
```

Arguments

n	how many colors you need
pal	col1~3; or a vector of colors, you can get from here too. <code>RColorBrewer::brewer.pal(5,"Set2")</code> <code>ggsci::pal_aaas()(5)</code>
picture	a picture file, colors will be extracted from the picture

Value

a vector of n colors

Examples

```
get_cols(10, "col2") -> my_cols
scales::show_col(my_cols)

scales::show_col(get_cols(15, RColorBrewer::brewer.pal(5, "Set2")))
scales::show_col(get_cols(15, ggsci::pal_aaas()(5)))
```

get_doi	<i>Download supplemental materials according to a doi</i>
---------	---

Description

Download supplemental materials according to a doi

Usage

```
get_doi(
  doi,
  dir = "~/Downloads/",
  bget_path = "~/software/bget_0.3.2_Darwin_64-bit/bget"
)
```

Arguments

doi	doi
dir	dir
bget_path	your bget_path

Value

file at work directory

gghuan	<i>Plot a doughnut chart</i>
--------	------------------------------

Description

Plot a doughnut chart

Usage

```
gghuan(
  tab,
  reorder = TRUE,
  mode = "1",
  topN = 5,
  name = TRUE,
  percentage = TRUE
)
```

Arguments

tab	two columns: first is type, second is number
reorder	reorder by number?
mode	plot style, 1~3
topN	plot how many top items
name	label the name
percentage	label the percentage

Value

a ggplot

Examples

```
a <- data.frame(type = letters[1:6], num = c(1, 3, 3, 4, 5, 10))
gghuan(a) + ggplot2::scale_fill_manual(values = get_cols(6, "col3"))
b <- data.frame(type = letters[1:12], num = c(1, 3, 3, 4, 15, 10, 35, 2:6))
gghuan(b) + ggplot2::theme(legend.position = "right")
```

gghuan2 *gghuan2 for multi-doughnut chart*

Description

gghuan2 for multi-doughnut chart

Usage

```
gghuan2(  
  tab = NULL,  
  `break` = 0.2,  
  name = TRUE,  
  number = TRUE,  
  percentage = FALSE,  
  text_col = "black"  
)
```

Arguments

tab	a dataframe with hierarchical structure
break	default 0.2
name	label the name
number	label the number
percentage	label the percentage
text_col	defalut, black

Value

a ggplot

Examples

```
data.frame(a = c("a", "a", "b", "b", "c"), aa = rep("a", 5),  
  b = c("a", LETTERS[2:5]), c = 1:5) %>% gghuan2()
```

```
grepl.data.frame
```

Grepl applied on a data.frame

Description

Grepl applied on a data.frame

Usage

```
grepl.data.frame(pattern, x, ...)
```

Arguments

pattern	search pattern
x	your data.frame
...	additional arguments for gerpl()

Value

a logical data.frame

Examples

```
matrix(letters[1:6], 2, 3) |> as.data.frame() -> a
grepl.data.frame("c", a)
grepl.data.frame("\\w", a)
```

```
group_box
```

Plot a boxplot

Description

Plot a boxplot

Usage

```
group_box(
  tab,
  group = NULL,
  metadata = NULL,
  mode = 1,
  group_order = NULL,
  facet_order = NULL,
  alpha = FALSE,
  method = "wilcox",
```

```

alpha_param = list(color = "red"),
p_value1 = FALSE,
p_value2 = FALSE,
stat_compare_means_param = NULL,
trend_line = FALSE,
trend_line_param = list(color = "blue")
)

```

Arguments

tab	your dataframe
group	which colname choose for group or a vector
metadata	the dataframe contains the group
mode	1~3, plot style
group_order	the order of x group
facet_order	the order of the facet
alpha	whether plot a group alphabeta by test of method
method	test method:wilcox, tukeyHSD, LSD, (default: wilcox), see multitest
alpha_param	parameters parse to geom_text
p_value1	multi-test of all group
p_value2	two-test of each pair
stat_compare_means_param	parameters parse to stat_compare_means
trend_line	add a trend line
trend_line_param	parameters parse to geom_smooth

Value

a ggplot

Examples

```

a <- data.frame(a = 1:18, b = runif(18, 0, 5))
group_box(a, group = rep(c("a", "b", "c"), each = 6))

```

hebing

Group your data

Description

Group your data

Usage

```
hebing(otutab, group, margin = 2, act = "mean")
```

Arguments

otutab	data.frame
group	group vector
margin	1 for row and 2 for column(default: 2)
act	do (default: mean)

Value

data.frame

Examples

```
data(otutab)
hebing(otutab, metadata$Group)
```

how_to_update_parameters

How to update parameters

Description

How to update parameters

Usage

```
how_to_update_parameters()
```

Value

No return value

how_to_use_parallel *How to use parallel*

Description

How to use parallel

Usage

```
how_to_use_parallel()
```

Value

No return value

is.ggplot.color *Judge if a characteristic is Rcolor*

Description

Judge if a characteristic is Rcolor

Usage

```
is.ggplot.color(color)
```

Arguments

color characteristic

Value

TRUE or FALSE

Examples

```
is.ggplot.color("red")  
is.ggplot.color("notcolor")  
is.ggplot.color(NA)  
is.ggplot.color("#000")
```

legend_size	<i>Scale a legend size</i>
-------------	----------------------------

Description

Scale a legend size

Usage

```
legend_size(scale = 1)
```

Arguments

scale	default: 1.
-------	-------------

Value

"theme" "gg"

lib_ps	<i>Attach packages or install packages have not benn installed</i>
--------	--

Description

Attach packages or install packages have not benn installed

Usage

```
lib_ps(p_list, ..., all_yes = FALSE, library = TRUE)
```

Arguments

p_list	a vector of packages list
...	packages
all_yes	all install try set to yes?
library	should library the package or just get Namespace ?

Value

No return value

little_guodong	<i>My cat.</i>
----------------	----------------

Description

my little cat named Guo Dong which drawn by my girlfriend.

Format

rastergrob object.

make_project	<i>Make a R-analysis project</i>
--------------	----------------------------------

Description

Make a R-analysis project

Usage

```
make_project(  
  pro_n,  
  root_dir = "~/Documents/R/",  
  bib = "~/Documents/R/pc_blog/content/bib/My Library.bib",  
  csl = "~/Documents/R/pc_blog/content/bib/science.csl"  
)
```

Arguments

pro_n	project name
root_dir	root directory
bib	cite papers bib, from Zotero
csl	cite papers format, default science.csl

Value

No return value

metadata	<i>test data for pcutils package.</i>
----------	---------------------------------------

Description

an otutab, metadata and a taxonomy table.

Format

contains an otutab, metadata and a taxonomy table.

otutab contains otutable rawdata

metadata contains metadata

taxonomy contains taxonomy table

mmscale	<i>Min_Max scale</i>
---------	----------------------

Description

Min_Max scale

Usage

```
mmscale(x, min_s = 0, max_s = 1, n = 1, plot = FALSE)
```

Arguments

x	a numeric vector
min_s	scale min
max_s	scale max
n	linear transfer for n=1; the slope will change if n>1 or n<1
plot	whether plot the transfer?

Value

a numeric vector

Examples

```
x <- runif(10)
mmscale(x, 5, 10)
```

multitest	<i>Multi-groups test</i>
-----------	--------------------------

Description

anova (parametric) and kruskal.test (non-parametric). Perform one-way ANOVA test comparing multiple groups. LSD and TukeyHSD are post hoc test of anova. dunn and nemenyi are post hoc test of kruskal.test. ttest or wilcox is just perform wilcox-test in each two group (no p.adjust).

Usage

```
multitest(var, group, print = TRUE, return = FALSE)
```

Arguments

var	numeric vector
group	more than two-levels group vector
print	whether print the result
return	return which method result (tukeyHSD or LSD or wilcox?)

Value

No value or a dataframe.

Examples

```
multitest(runif(30), rep(c("a", "b", "c"), each = 10), print = FALSE, return = "wilcox") -> aa
```

my_cat	<i>Show my little cat named Guo Dong which drawn by my girlfriend.</i>
--------	--

Description

Show my little cat named Guo Dong which drawn by my girlfriend.

Usage

```
my_cat(mode = 1)
```

Arguments

mode	1~2
------	-----

Value

a ggplot

my_circo	<i>My circo plot</i>
----------	----------------------

Description

My circo plot

Usage

```
my_circo(
  df,
  reorder = TRUE,
  pal = NULL,
  mode = c("circlize", "chorddiag"),
  ...
)
```

Arguments

df	dataframe with three column
reorder	reorder by number?
pal	a vector of colors, you can get from here too. <code>RColorBrewer::brewer.pal(5,"Set2")</code> <code>ggsci::pal_aaas()(5)</code>
mode	"circlize","chorddiag"
...	chordDiagram

Value

chordDiagram

Examples

```
data.frame(a=c("a","a","b","b","c"),b=c("a",LETTERS[2:5]),c=1:5)%>%my_circo(mode="circlize")
```

my_lm	<i>Fit a linear model and plot</i>
-------	------------------------------------

Description

Fit a linear model and plot

Usage

```
my_lm(tab, var, metadata = NULL, ...)
```

Arguments

tab	your dataframe
var	which colname choose for var or a vector
metadata	the dataframe contains the var
...	parameters parse to <code>geom_point</code>

Value

a ggplot

Examples

```
my_lm(runif(50), var = 1:50)
my_lm(c(1:50) + runif(50, 0, 5), var = 1:50)
```

otutab	<i>test data for pcutils package.</i>
--------	---------------------------------------

Description

an otutab, metadata and a taxonomy table.

Format

contains an otutab, metadata and a taxonomy table.

otutab contians otutable rawdata

metadata contians metadata

taxonomy contians taxonomy table

plotgif	<i>Plot a gif</i>
---------	-------------------

Description

Plot a gif

Usage

```
plotgif(plist, file, mode = "gif")
```

Arguments

plist	plot list
file	prefix of your .gif file
mode	"gif" or "html"

Value

No return value

plotpdf	<i>Plot a multi-pages pdf</i>
---------	-------------------------------

Description

Plot a multi-pages pdf

Usage

```
plotpdf(
  plist,
  file,
  width = 8,
  height = 7,
  brower = "/Applications/Microsoft Edge.app/Contents/MacOS/Microsoft Edge",
  ...
)
```

Arguments

plist	plot list
file	prefix of your .pdf file
width	width
height	height
brower	the path of Google Chrome, Microsoft Edge or Chromium in your computer.
...	additional arguments

Value

No return value

read.file	<i>Read some special format file</i>
-----------	--------------------------------------

Description

Read some special format file

Usage

```
read.file(file, format = NULL, just_print = FALSE)
```

Arguments

file	file path
format	"blast","diamond"
just_print	just print the file

Value

data.frame

read_fasta	<i>Read fasta file</i>
------------	------------------------

Description

Read fasta file

Usage

```
read_fasta(fasta_file)
```

Arguments

fasta_file	file path
------------	-----------

remove.outliers	<i>Remove outliers</i>
-----------------	------------------------

Description

Remove outliers

Usage

```
remove.outliers(x, factor = 1.5)
```

Arguments

x	a numeric vector
factor	default 1.5

Value

a numeric vector

Examples

```
remove.outliers(c(1, 10:15))
```

rgb2code	<i>Transform a rgb vector to a Rcolor code</i>
----------	--

Description

Transform a rgb vector to a Rcolor code

Usage

```
rgb2code(x, rev = FALSE)
```

Arguments

x	vector or three columns data.frame
rev	reverse, transform a Rcolor code to a rgb vector

Value

Rcolor code like "#69C404"

Examples

```
rgb2code(c(12, 23, 34))  
rgb2code("#69C404", rev = TRUE)
```

sanxian	<i>Three-line table</i>
---------	-------------------------

Description

Three-line table

Usage

```
sanxian(df, digits = 3, nrow = 10, ncol = 10, fig = FALSE, ...)
```

Arguments

df	a data.frame
digits	how many digits should remain
nrow	show how many rows
ncol	show how many columns
fig	output as a figure
...	additional arguments e.g.(rows=NULL)

Value

a ggplot

Examples

```
data(otutab)  
sanxian(otutab)
```

stackplot	<i>Plot a stack plot</i>
-----------	--------------------------

Description

Plot a stack plot

Usage

```

stackplot(
  otutab,
  metadata = NULL,
  group = "Group",
  get_data = FALSE,
  bar_params = list(width = 0.7, position = "stack"),
  topN = 8,
  others = TRUE,
  relative = TRUE,
  legend_title = "",
  stack_order = TRUE,
  group_order = FALSE,
  facet_order = FALSE,
  style = c("group", "sample")[1],
  flow = FALSE,
  flow_params = list(lode.guidance = "frontback", color = "darkgray"),
  number = FALSE,
  format_params = list(digits = 2),
  text_params = list(position = position_stack())
)

```

Arguments

otutab	otutab
metadata	metadata
group	one group name of columns of metadata
get_data	just get the formatted data?
bar_params	parameters parse to geom_bar
topN	plot how many top species
others	should plot others?
relative	transfer to relative or absolute
legend_title	fill legend_title
stack_order	the order of stack fill
group_order	the order of x group
facet_order	the order of the facet
style	"group" or "sample"
flow	should plot a flow plot?
flow_params	parameters parse to geom_flow
number	show the number?
format_params	parameters parse to format
text_params	parameters parse to geom_text

Value

a ggplot

Examples

```
data(otutab)
stackplot(otutab, metadata, group = "Group")

stackplot(otutab, metadata, group = "Group", group_order = TRUE, flow = TRUE, relative = FALSE)
```

strsplit2	<i>Split Composite Names</i>
-----------	------------------------------

Description

Split Composite Names

Usage

```
strsplit2(x, split, colnames = NULL, ...)
```

Arguments

x	character vector
split	character to split each element of vector on, see strsplit
colnames	colnames for the result
...	other arguments are passed to strsplit

Value

data.frame

Examples

```
strsplit2(c("a;b", "c;d"), ";")
```

taxonomy	<i>test data for pcutils package.</i>
----------	---------------------------------------

Description

an otutab, metadata and a taxonomy table.

Format

contains an otutab, metadata and a taxonomy table.

otutab contains otutable rawdata

metadata contains metadata

taxonomy contains taxonomy table

tax_pie	<i>Pie plot</i>
---------	-----------------

Description

Pie plot

Usage

```
tax_pie(otutab, topN = 6, ...)
```

Arguments

otutab	otutab
topN	topN
...	add

Value

a ggplot

Examples

```
data(otutab)
tax_pie(otutab, topN = 7)
```

tax_wordcloud	<i>Word cloud plot</i>
---------------	------------------------

Description

Word cloud plot

Usage

```
tax_wordcloud(str_vector)
```

Arguments

str_vector string vector

Value

a htmlwidget

Examples

```
data(otutab)
tax_wordcloud(taxonomy$Genus)
```

trans_format	<i>Transfer the format of file</i>
--------------	------------------------------------

Description

Transfer the format of file

Usage

```
trans_format(
  file,
  to_format,
  format = NULL,
  ...,
  browser = "/Applications/Microsoft Edge.app/Contents/MacOS/Microsoft Edge"
)
```

Arguments

file	input file
to_format	transfer to
format	input file format
...	additional argument
brower	the path of Google Chrome, Microsoft Edge or Chromium in your computer.

Value

file at work directory

twotest	<i>Two-group test</i>
---------	-----------------------

Description

Two-group test

Usage

```
twotest(var, group)
```

Arguments

var	numeric vector
group	two-levels group vector

Value

No return value

Examples

```
twotest(runif(20), rep(c("a", "b"), each = 10))
```

update_param	<i>Update the parameters</i>
--------------	------------------------------

Description

Keep the different parameters while use the same name in update first.

Usage

```
update_param(default, update)
```

Arguments

default	default (data.frame, list, vector)
update	update (data.frame, list, vector)

Value

same class of your input (data.frame, list or vector)

Examples

```
update_param(list(a=1,b=2),list(b=5,c=5))
```

venn	<i>Plot a general venn (upset, flower)</i>
------	--

Description

Plot a general venn (upset, flower)

Usage

```
venn(...)

## S3 method for class 'list'
venn(aa, mode = "venn", ...)

## S3 method for class 'data.frame'
venn(otutab, mode = "venn", ...)
```

Arguments

...	add
aa	list
mode	"venn","venn2","upset","flower"
otutab	table

Value

a plot
a plot
a plot

Examples

```
aa <- list(a = 1:3, b = 3:7, c = 2:4)
venn(aa, mode = "venn")
venn(aa, mode = "venn2", type = "ChowRuskey")
venn(aa, mode = "upset")
data(otutab)
venn(otutab, mode = "flower")
```

write_fasta

Write a data.frame to fasta

Description

Write a data.frame to fasta

Usage

```
write_fasta(df, file_path, str_per_line = 70)
```

Arguments

df	data.frame
file_path	output file path
str_per_line	how many base or animo acid in one line, if NULL, one sequence in one line.

Value

file

Index

add_alpha, 3
add_analysis, 3
add_theme, 4

change_fac_lev, 5
china_map, 5
chordDiagram, 20
copy_vector, 6
count2, 6

dabiao, 7
del_ps, 7

explode, 8

fittest, 8
format, 26

geom_bar, 26
geom_flow, 26
geom_point, 21
geom_smooth, 13
geom_text, 13, 26
get_cols, 9
get_doi, 9
gghuan, 10
gghuan2, 11
grepl.data.frame, 12
group_box, 12

hebing, 14
how_to_update_parameters, 14
how_to_use_parallel, 15

is.ggplot.color, 15

legend_size, 16
lib_ps, 16
little_guodong, 17

make_project, 17

metadata, 18
mmscale, 18
multitest, 13, 19
my_cat, 19
my_circo, 20
my_lm, 20

otutab, 21

plotgif, 21
plotpdf, 22

read.file, 23
read_fasta, 23
remove.outliers, 24
rgb2code, 24

sanxian, 25
stackplot, 25
stat_compare_means, 13
strsplit, 27
strsplit2, 27

tax_pie, 28
tax_wordcloud, 29
taxonomy, 28
trans_format, 29
twotest, 30

update_param, 31

venn, 31

write_fasta, 32